



10/02/00

IN THE U.S. PATENT AND TRADEMARK OFFICE
Patent Application Transmittal Letter

10/02/00

10/02/00

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Sir:

Transmitted herewith for filing under 37 CFR 1.53(b) is a(n): ☒ Utility ☐ Design

- ☒ original patent application,
☐ continuation-in-part application

INVENTOR(S): David BOHAN

TITLE: UTILIZATION OF THIRD PARTY LEGACY DATA LIST

Enclosed are:

- ☒ The Declaration and Power of Attorney. ☒ signed ☐ unsigned or partially signed
☒ 6 sheets of drawings (one set) ☐ Associate Power of Attorney
☐ Form PTO-1449 ☐ Information Disclosure Statement and Form PTO-1449
☐ Priority document(s) ☐ (Other) (fee \$)

| CLAIMS AS FILED BY OTHER THAN A SMALL ENTITY | | | | |
|--|---------------------|---------------------|-------------|---------------|
| (1) FOR | (2) NUMBER FILED | (3) NUMBER EXTRA | (4) RATE | (5) TOTALS |
| TOTAL CLAIMS | 20 — 20 | 0 | X \$18 | \$ 0 |
| INDEPENDENT CLAIMS | 3 — 3 | 0 | X \$78 | \$ 0 |
| ANY MULTIPLE DEPENDENT CLAIMS | 0 | | \$260 | \$ 0 |
| BASIC FEE: Design (\$310.00); Utility (\$690.00) | | | | \$ 690 |
| TOTAL FILING FEE | | | | \$ 690 |
| OTHER FEES | | | | \$ |
| TOTAL CHARGES TO DEPOSIT ACCOUNT | | | | \$ 690 |

Charge \$ 690 to Deposit Account 08-2025. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16, 1.17, 1.19, 1.20 and 1.21. A duplicate copy of this sheet is enclosed.

"Express Mail" label no. _____
Date of Deposit _____

I hereby certify that this is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231.

By _____
Typed Name:

Respectfully submitted,

David BOHAN

By

Ki S. Kim

Attorney/Agent for Applicant(s)

Reg. No. 43,450

Date: 10/2/2000

Telephone No.: (202) 663-6475

UTILIZATION OF THIRD PARTY LEGACY DATA LIST

FIELD OF THE INVENTION

This invention relates generally to computer software, and more particularly to methods of allow use of legacy data lists by a newly installed application program.

BACKGROUND OF THE INVENTION

Data lists are used extensively in various application programs. For example, a phonebook is typically a requirement for computer facsimile software. As another example, e-mail address books and contact information lists are commonly incorporated into e-mail programs. Typically, the data list is maintained in a custom format for its particular application program, and not compatible with other application programs.

When a new application program is installed to replace an existing, or a legacy, application program, a new user must populate the data list associated with the new application program. Such population can be done manually or semi-automatically by use of an import function. In the latter case, a phonebook or address book, for example, from the legacy data list from the legacy application program is imported into the new application problem. A problem with either of the above technique is that two separate data lists must be maintained and synchronized. This requires extra and duplicated effort on the part of the user, and is thus undesirable.

Thus, there is a need for a method and apparatus for allowing a new application program to utilize data lists of one or more legacy application program without requiring a user to maintain and/or synchronize two separate data lists.

SUMMARY OF THE INVENTION

In accordance with the principles of the present invention, a method of providing an access to one or more third party legacy data list to a user of an application program of a computer system comprises querying an operating system, by the application program upon start of the application program, whether one or more plug-in module is registered in a registry of an operating system, the one or more plug-in modules being capable of interfacing

1 with corresponding respective ones of the one or more third party legacy data list, receiving,
2 from the one or more plug-in modules found in the registry, identifications of ones of the one
3 or more third party legacy data list corresponding to the found one or more plug-in modules,
4 and providing a list of the identifications to the user through a user interface of the application
5 program.

6 In accordance with another aspect of the present invention, a system for providing an
7 access to one or more third party legacy data list to a user of a computer system comprises
8 one or more plug-in module, each of which being capable of interfacing with an associated
9 one of the one or more third party data list, each of the one or more plug-in modules being
10 registered in a registry of an operating system of the computer system, and an application
11 program having a user interface, the application program upon starting being in
12 communication with the operating system to query the registry to determine registered ones
13 of the one or more plug-in module, the application program further configured to query each
14 of the registered ones of the one or more plug-in modules for names of the one or more third
15 party legacy data list, the application program being configured to provide a list of the names
16 of the one or more third party legacy data list to the user through the user interface.

17 In accordance with yet another aspect of the present invention, a computer readable
18 storage medium on which is embedded one or more computer programs, the one or more
19 computer programs implementing a method of providing an access to one or more third party
20 legacy data list to a user of an application program of a computer system, the one or more
21 computer programs comprises a set of instructions for querying an operating system, by the
22 application program upon start of the application program, whether one or more plug-in
23 module is registered in a registry of an operating system, the one or more plug-in modules
24 being capable of interfacing with corresponding respective ones of the one or more third party
25 legacy data list, receiving, from the one or more plug-in modules found in the registry,
26 identifications of ones of the one or more third party legacy data list corresponding to the
27 found one or more plug-in modules, and providing a list of the identifications to the user
28 through a user interface of the application program.

29 In comparison to known prior art, certain embodiments of the invention are capable of
30 achieving certain advantages, including some or all of the following: (1) saving time and

effort of the user; (2) eliminating the need for a user to populate a new data list; (3) eliminating the need to maintain and synchronize multiple data lists; (4) avoiding errors that can be introduced during the population process; and (5) eliminating the need for a user to learn a new user interface. Those skilled in the art will appreciate these and other advantages and benefits of various embodiments of the invention upon reading the following detailed description of a preferred embodiment with reference to the below-listed drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Features and advantages of the present invention will become apparent to those skilled in the art from the following description with reference to the drawings, in which:

Figure 1 shows an exemplary embodiment of the relevant portions of a system for providing user access of third party legacy data list in accordance with the principles of the present invention;

Figure 1A shows a more detailed depiction of an exemplary embodiment of one of the plug-in module shown in Figure 1;

Figure 2 shows a flowchart of an exemplary embodiment of the application program installation process in accordance with the principles of the present invention;

Figure 3 shows a flowchart of an exemplary embodiment of the plug-in modules discovery process in accordance with the principles of the present invention;

Figure 4 shows an illustrative exemplary embodiment of the user interface for the application program in accordance with the principles of the present invention;

Figure 5 shows a flowchart of an exemplary embodiment of the user selection and access of the third party legacy data list process in accordance with the principles of the present invention; and

Figure 6 shows a flowchart of an exemplary embodiment of the application update process in accordance with the principles of the present invention;

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

For simplicity and illustrative purposes, the principles of the present invention are described by referring mainly to an exemplar embodiment thereof, particularly with

1 references to an example of a facsimile software. However, one of ordinary skill in the art
2 would readily recognize that the same principles are equally applicable to, and can be
3 implemented in, any device configuration setting of any other types of application programs
4 that utilizes a data list, and that any such variation would be within such modifications that do
5 not depart from the true spirit and scope of the present invention.

6 In accordance with the principles of the present invention, a method and system of a
7 computer application program allows a user of the application program to utilize third party
8 legacy data lists, e.g., contact information lists, telephone and facsimile numbers, scheduling
9 and/or appointment information or the like, through a common interface, without requiring
10 the user to manually type in the data into a new data list or to maintain multiple data lists.
11 For each of the third party legacy data list installed on the computer, one of a plurality of
12 plug-in modules also installed on the computer acts as an interface between the application
13 program and the legacy data list. Each of the plug-in module is an object accessible by the
14 application program at run-time.

15 In one embodiment, the plug-in modules are compiled and a dynamic linked library
16 (DLL) and registered in the operating system registry based on an object oriented
17 specification, e.g., the component object model (COM) specification provided by the
18 Microsoft Corporation of Redmond, Washington, USA. The plug-in modules are listed under
19 a COM category in the registry of the operating system, and are discovered by the application
20 program at start up. Additional plug-in modules may be added at any time simply by
21 registering the additional plug-in modules. The next time the application program is started,
22 the additional plug-in modules are automatically discovered. Using the discovered plug-in
23 modules, the application program allows the user to access and/or edit the third party legacy
24 data lists through the user interface of the application program. In an alternative embodiment,
25 upon an indication, by the user of the application program, a desire to edit data of a particular
26 third party legacy data list, the corresponding plug-in module causes the editor user interface
27 screen of the legacy data list to be displayed to the user. The user is allowed to edit the data
28 list using the user interface of the legacy data list.

29 In particular, Figure 1 shows an exemplary embodiment of the relevant portions of a
30 system 100 for providing user access of third party legacy data list in accordance with the

1 principles of the present invention. The system **100** comprises an operating system (OS) **101**
2 installed on the computer (not shown). The OS **101** may be, for example, the WINDOWSTM
3 sold by the Microsoft Corporation of Redmond, Washington, USA. The operating system
4 includes a component registry **102** that keeps a list of system components and resources
5 required by an application program **103** installed on the computer. The application program
6 **103** may comprise any program that is require to or is desirable to keep a data list. For
7 example, a facsimile software may keep a list of names and facsimile telephone numbers of
8 many potential recipients. An e-mail software may need to keep a list of e-mail addresses of
9 various e-mail users. Many personal organizer programs keep contact information,
10 scheduling/appointment lists, or the like.

11 The application program **103** may also comprise a user interface **104** for providing a
12 user of the application program means for interact with the application program **103**. In
13 accordance with the principles of the present invention, the user interface **104** serves as a
14 common interface through which the user may access any number of third party legacy data
15 lists **106**, i.e., the legacy data list #1 through legacy data list #N (where N may be any integer
16 greater than equal to 1) installed on the computer.

17 The system **100** may also comprise any number of plug-in modules (PM) **105**, one of
18 which is shown in more detail in Fig. 1A. As shown, each plug-in module **105** comprises an
19 application program interface portion **105A** and a third party legacy (TPL) data list interface
20 portion **105B**. The application program interface portions **105A** of all plug-in modules are
21 identical to each other. Thus, the application program **103** may use one common protocol
22 and/or function call set to communicate with all plug-in modules **105**. Thus, when an
23 additional plug-in module is installed on the computer after the installation of the application
24 program **103**, the application program **103** need not be recompiled or modified in any way.
25 As will described in more detail later, the newly added plug-in modules are discovered by the
26 application program **103**, and are used by the application in the same manner that the existing
27 plug-in modules are used.

28 In accordance with an embodiment of the present invention, each of the plug-in
29 modules **105** is compiled as a dynamic link library, with which the application program **103**
30 may communicate during run-time. In a preferred embodiment of the present invention, the

1 application program **103** and each of the plug-in modules **105** are made to be in compliance
2 with the component object model (COM) specification included in the WINDOWS™
3 operating system sold by the Microsoft Corporation of Redmond, Washington, USA.

4 Referring again to Fig. 1A, the TPL data list interface portion **105B** of each of the
5 plug-in modules is developed specifically for the third party legacy data list the particular
6 plug-in module is intended to support. The information, e.g., the function call sets, necessary
7 to interface to and to access a third party legacy data list is obtained from the information
8 typically published by the supplier of the data list.

9 Each of the plug-in modules perform any number of the following tasks: 1) return the
10 name of the particular data list it supports (or the name of the legacy application program
11 from which the particular data list was created) in response to a name request function call
12 from the application program; 2) determine whether the data list it supports is installed on the
13 computer in response to an availability check function call from the application program; 3)
14 display a editor user interface (of either the application program **103** or the legacy data list
15 **106**), through which a user can add, delete or modify any datum of the particular data list it
16 supports in response to a edit request function call from the application program; 4) allow
17 creation of a distribution (or recipient) list in response to a group creation function call from
18 the application program; and 5) provide the data set, e.g., the list of telephone numbers, for
19 access by a user, in response to a data request function call from the application program.

20 Figure 2 shows a flowchart of an exemplary embodiment of the application program
21 installation process in accordance with the principles of the present invention;

22 As mentioned, each of the plug in modules are compiled as an runtime accessible
23 object, e.g., a WINDOWS™ Dynamic Link Library (DLL), particularly as a component
24 object module (COM) object, or a system object model (SOM) promulgated by the IBM
25 Corporation of Armonk, NY, U.S.A., or the like.

26 The application program installation process **200** begins in step **201**, e.g., when a user
27 causes an installer program to be executed on the computer. In step **202**, the installer program
28 unpacks and copies the necessary program components as it is known to those familiar with
29 program installers. Then, in step **203**, an entry in the operating system registry **102** for each of
30 the plug-in modules. In a preferred embodiment of the present invention, the registration of

1 the plug-in modules comprise adding a COM category entry in the registry, e.g., COM
2 category: <Legacy Data List Plug-in Modules>, and listing each of the plug-in modules 105
3 as a COM object with a unique identifier under that COM category.

4 Referring now to Figure 3, an exemplary embodiment of the plug-in modules
5 discovery process in accordance with the principles of the present invention will be described.

6 Upon installation and/or start of the application program, the application program in
7 step 301, the application program sends a standard Windows function call, e.g., a COM
8 category call, to query the operating system for the list of plug-in modules under the
9 particular COM category, e.g., the <Legacy Data List Plug-in Modules> COM category, in
10 step 302.

11 In step 303, the operating system 101 returns the unique identifiers of the plug-in
12 modules registered in the registry 102. The application program 103 at this point knows how
13 many plug-in modules are installed on the computer. In step 304, the application program,
14 using the unique identifiers, sends a name request function call to each of the plug-in
15 modules, and receives the name of the data list from each of the plug-in modules. The
16 application program 103, then sends an availability check function call to each of the plug-in
17 modules in step 305. The application program 103 makes a determination, in step 306,
18 whether any of the plug-in modules has returned with a response that indicates the
19 corresponding data list is installed on the computer. If no plug-in module indicates that the
20 data list it supports is installed on the computer, the application program 103 may ask the user
21 of the application program whether the user wishes to create a new data list in step 308, and if
22 the user so wishes, may provide a user interface screen (not shown) to allow the user to
23 manually enter new data to create a new data list in step 309. The application program 103
24 then waits for a user action in step 310.

25 If, on the other hand, it is determined (in step 306) that one or more data list is
26 installed on the computer, the application program 103 causes the names of all data lists that
27 are installed on the computer to be displayed to the user through the user interface 104 to
28 allow the user to select a desired data list from the listed data lists.

29 As can be appreciated from the above description, once a plug-in module is registered
30 with the operating system, the application program may discover the plug-in module and

utilize the same without any modification to the application program **103**.

Figure 4 shows an illustrative exemplary embodiment of the user interface for the application program **103**, e.g., as implemented as a facsimile software, in accordance with the principles of the present invention. The facsimile software user interface **400** comprises, in addition to the various input windows and selection buttons to carry out the functionality of the facsimile software, a drop down selection box **401**, which lists the legacy data lists, e.g., in this example phone books from various legacy facsimile software, installed on the computer as identified by the application program according to the process described above.

In Fig. 4, it is shown that the user has selected, e.g., the legacy data list #2. The data set of the legacy data list #2 appears in the display window **402**. The user is also allowed to edit the data set of the selected data list by selecting the "Edit" selection button **403**, or to create a recipient list by selecting the "Create Recipient List" selection button **404**. When the user selects the "Edit" selection button **403**, in an embodiment of the present invention, the plug-in module **105** of the selected legacy data list **106** causes an edit user interface of the legacy data list **106** to be displayed to the user, and allows the user to add, delete, copy and/or modify the data of the legacy data list using a user interface the user may be familiar. In an alternative embodiment, the plug-in module **105** may cause an edit user interface of the application program **103** to be provided to the user.

In a particular embodiment of the present invention, the selected data list is stored as the "default data source", and is automatically selected the next time the application program **103** is run. The user can use the drop down box **402** to choose another source of data.

Referring to Figure 5, an exemplary embodiment of the user selection and access of the third party legacy data list process **500** in accordance with the principles of the present invention will now be described. When a user selects one of the data lists, e.g., from the drop down selection box **401**, the application program detects the selection in step **501**. The application program **103** then sends a data request function call to the plug-in module corresponding to the selected data list in step **502**.

The corresponding plug-in module **105** obtains the data set from the selected data list using the TPL data list interface portion **105B**, and returns the data set to the application

1 program 103 in step 502. In step 504, the data set is then displayed to the user, e.g., in the
2 display window 402, from which the user can access the data.

3 If, in step 505, the application program 103 detects that the user wishes to modify any
4 datum in the selected third party legacy data list 105, e.g., when the "Edit" selection button
5 403 is selected, the application program 103 sends an edit request function call to the plug-in
6 module supporting the selected data list in step 506. The plug-in module then causes an edit
7 user interface screen (not shown) to be displayed in step 507, and the user is allowed to add,
8 delete and/or modify any datum from the data set of the selected data list. The edit user
9 interface screen may be either a user interface screen of the application program 103 or a user
10 interface screen of the legacy data list 106. The process then ends in step 509 until another
11 user action is detected in step 501, in which case, the process 500 is repeated.

12 Figure 6 shows a flowchart of an exemplary embodiment of the application update
13 process in accordance with the principles of the present invention. As previously mentioned,
14 even after the application program 103 is installed on a field computer, when a plug-in
15 module for a previously unsupported data list is desired, the plug-in may be developed and
16 delivered to the field, or downloaded from the Internet by the user, and installed on the field
17 computer.

18 The installer program for the new plug-in module may unpack and copy the new plug-
19 in module(s) to a memory storage of the computer, e.g., a hard disk, in step 602. In step 603,
20 the installer then registers the new plug-in module with the operating system in a similar
21 manner as previously described in connection with Fig. 2. Once the new plug-in module is
22 registered, the application program 103 will automatically discover the new plug-in module
23 the next time the application program is run as described above in Fig. 3.

24 What has been described and illustrated herein is a preferred embodiment of the
25 invention along with some of its variations. The terms, descriptions and figures used herein
26 are set forth by way of illustration only and are not meant as limitations. Those skilled in the
27 art will recognize that many variations are possible within the spirit and scope of the
28 invention, which is intended to be defined by the following claims -- and their equivalents --
29 in which all terms are meant in their broadest reasonable sense unless otherwise indicated.

CLAIMS

What is claimed is:

1. A method of providing an access to one or more third party legacy data list to a user of an application program of a computer system, comprising:

querying an operating system, by said application program upon start of said application program, whether one or more plug-in module is registered in a registry of an operating system, said one or more plug-in modules being capable of interfacing with corresponding respective ones of said one or more third party legacy data list;

receiving, from said one or more plug-in modules found in said registry, identifications of ones of said one or more third party legacy data list corresponding to said found one or more plug-in modules; and

providing a list of said identifications to said user through a user interface of said application program.

2. The method of providing an access to one or more third party legacy data list in accordance with claim 1, further comprising:

adding to said computer system one or more additional plug-in module capable of interfacing with one or more additional third party legacy data list; and

registering said one or more additional plug-in modules in said registry of said operating system, said application program being configured to find said one or more additional plug-in modules when said application program is started after said addition of said one or more additional plug-in modules.

3. The method of providing an access to one or more third party legacy data list in accordance with claim 1, further comprising:

allowing said user to select a selected one of said one or more third party legacy data list from said list of identifications; and

allowing said user to edit at least one datum of said selected one of said one or more third party legacy data list through an edit user interface of said selected one of said one or more third party legacy data list, said application program communicating with said selected

one of said one or more third party legacy data list through corresponding one of said one or more plug-in module.

4. The method of providing an access to one or more third party legacy data list in accordance with claim 1, further comprising:

allowing said user to select a selected one of said one or more third party legacy data list from said list of identifications; and

allowing said user to access at least one datum of said selected one of said one or more third party legacy data list through said user interface of said application program, said application program communicating with said selected to said one of said one or more third party legacy data list through corresponding one of said one or more plug-in module.

5. The method of providing an access to one or more third party legacy data list in accordance with claim 4, wherein:

said application program comprises a facsimile software; and

wherein said one or more third party legacy data list comprise one or more list of names and telephone numbers stored using a legacy facsimile software.

6. The method of providing an access to one or more third party legacy data list in accordance with claim 4, wherein:

said application program comprises an e-mail software; and

wherein said one or more third party legacy data list comprise one or more list of names and e-mail addresses stored using a legacy e-mail software.

7. The method of providing an access to one or more third party legacy data list in accordance with claim 4, wherein:

said application program comprises a personal organizer software; and

wherein said one or more third party legacy data list comprise one or more contact information and appointment information stored using a legacy personal organizer software.

1 **8.** A system for providing an access to one or more third party legacy data list to a
2 user of a computer system, comprising:

3 one or more plug-in module, each of which being capable of interfacing with an
4 associated one of said one or more third party data list, each of said one or more plug-in
5 modules being registered in a registry of an operating system of said computer system; and

6 an application program having a user interface, said application program upon starting
7 being in communication with said operating system to query said registry to determine
8 registered ones of said one or more plug-in module, said application program further
9 configured to query each of said registered ones of said one or more plug-in modules for
10 names of said one or more third party legacy data list, said application program being
11 configured to provide a list of said names of said one or more third party legacy data list to
12 said user through said user interface.

13
14 **9.** The system for providing an access to one or more third party legacy data list
15 according to claim 8, further comprising:

16 an installer software configured to copy said one or more plug-in module to said
17 computer system, said installer software further configured to registering said one or more
18 copied plug-in modules in said registry of said operating system.

19
20 **10.** The system for providing an access to one or more third party legacy data list
21 according to claim 8, wherein:

22 said user interface is configured to allow said user to select a selected one of said one
23 or more third party legacy data list from said list of identifications.

24
25 **11.** The system for providing an access to one or more third party legacy data list
26 according to claim 10, wherein:

27 said application program comprises a facsimile software; and

28 wherein said one or more third party legacy data list comprise one or more list of
29 names and telephone numbers stored using a legacy facsimile software.

1 12. The system for providing an access to one or more third party legacy data list
2 according to claim 10, wherein:

3 said application program comprises an e-mail software; and
4 wherein said one or more third party legacy data list comprise one or more list of
5 names and e-mail addresses stored using a legacy e-mail software.
6

7 13. The system for providing an access to one or more third party legacy data list
8 according to claim 10, wherein:

9 said application program comprises a personal organizer software; and
10 wherein said one or more third party legacy data list comprise one or more contact
11 information and appointment information stored using a legacy personal organizer software.
12

13 14. A computer readable storage medium on which is embedded one or more
14 computer programs, said one or more computer programs implementing a method of
15 providing an access to one or more third party legacy data list to a user of an application
16 program of a computer system, said one or more computer programs comprising a set of
17 instructions for:

18 querying an operating system, by said application program upon start of said
19 application program, whether one or more plug-in module is registered in a registry of an
20 operating system, said one or more plug-in modules being capable of interfacing with
21 corresponding respective ones of said one or more third party legacy data list;

22 receiving, from said one or more plug-in modules found in said registry,
23 identifications of ones of said one or more third party legacy data list corresponding to said
24 found one or more plug-in modules; and

25 providing a list of said identifications to said user through a user interface of said
26 application program.
27
28

1 15. The computer readable storage medium in according to claim 14, wherein said one
2 or more computer programs further comprising a set of instructions for:
3 adding to said computer system one or more additional plug-in module capable of
4 interfacing with one or more additional third party legacy data list; and
5 registering said one or more additional plug-in modules in said registry of said
6 operating system, said application program being configured to find said one or more
7 additional plug-in modules when said application program is started after said addition of said
8 one or more additional plug-in modules.

9
10 16. The computer readable storage medium in according to claim 14, wherein said one
11 or more computer program further comprising a set of instructions for:
12 allowing said user to select a selected one of said one or more third party legacy data
13 list from said list of identifications; and
14 allowing said user to edit at least one datum of said selected one of said one or more
15 third party legacy data list through an edit user interface of said selected one of said one or
16 more third party legacy data list, said application program communicating with said selected
17 one of said one or more third party legacy data list through corresponding one of said one or
18 more plug-in module.

19
20 17. The computer readable storage medium in according to claim 14, wherein said one
21 or more computer program further comprising a set of instructions for:
22 allowing said user to select a selected one of said one or more third party legacy data
23 list from said list of identifications; and
24 allowing said user to access at least one datum of said selected one of said one or
25 more third party legacy data list through said user interface of said application program, said
26 application program communicating with said selected to said one of said one or more third
27 party legacy data list through corresponding one of said one or more plug-in module.

1 **18.** The computer readable storage medium in according to claim 17, wherein :
2 said application program comprises a facsimile software; and
3 wherein said one or more third party legacy data list comprise one or more list of
4 names and telephone numbers stored using a legacy facsimile software.
5

6 **19.** The computer readable storage medium in according to claim 17, wherein :
7 said application program comprises an e-mail software; and
8 wherein said one or more third party legacy data list comprise one or more list of
9 names and e-mail addresses stored using a legacy e-mail software.
10

11 **20.** The computer readable storage medium in according to claim 17, wherein :
12 said application program comprises a personal organizer software; and
13 wherein said one or more third party legacy data list comprise one or more contact
14 information and appointment information stored using a legacy personal organizer software.
15

[illegible]

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19

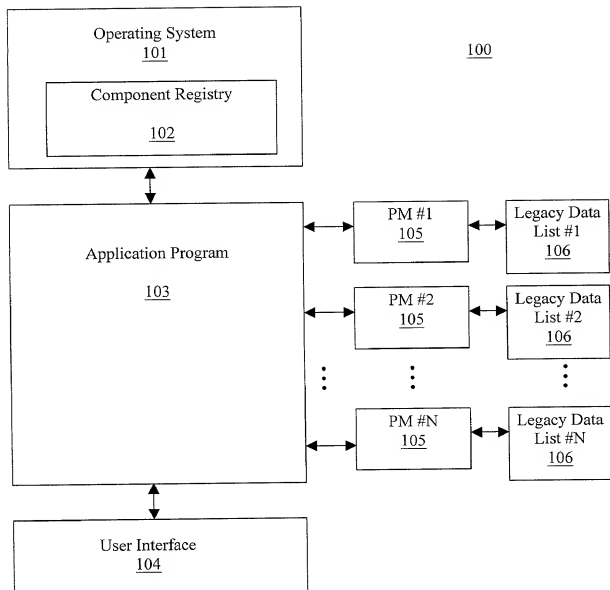


Fig. 1

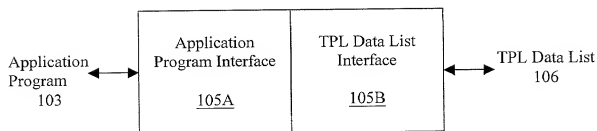


Fig. 1A

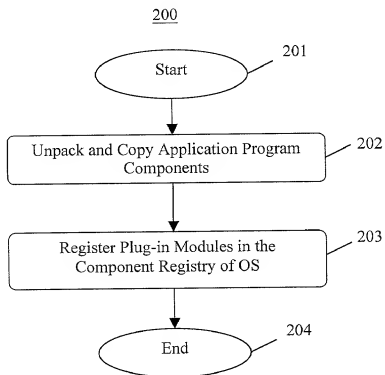


Fig. 2

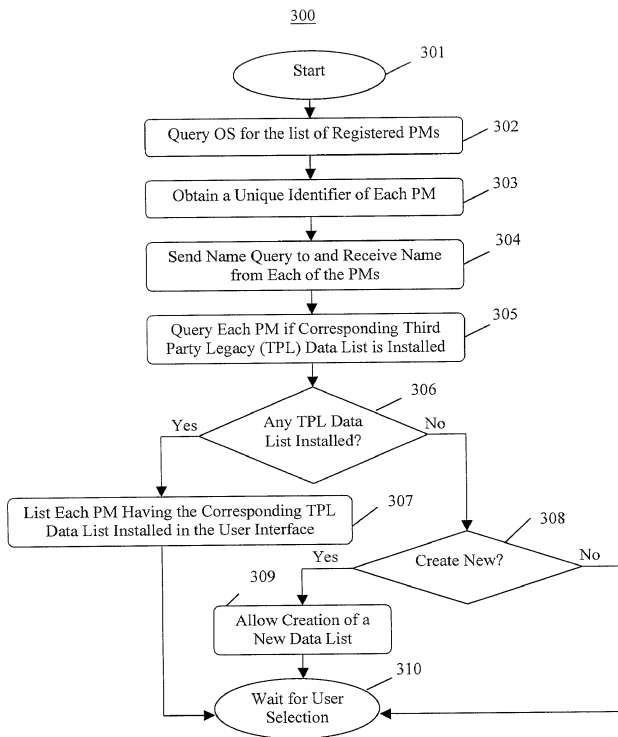


Fig. 3

500
501
502
503
504
505
506
507
508
509

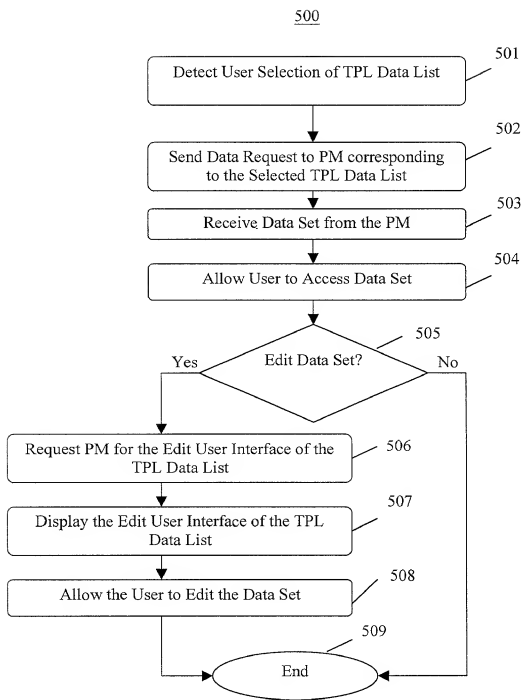


Fig. 5

600

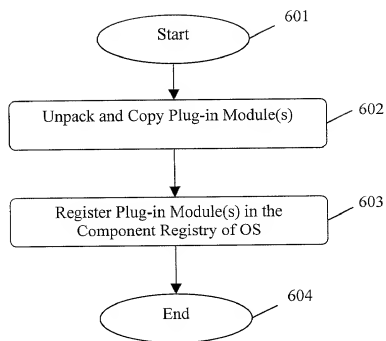


Fig. 6

**DECLARATION AND POWER OF ATTORNEY
FOR PATENT APPLICATION**ATTORNEY DOCKET NO. 10002960-1

As a below named inventor, I hereby declare that:

My residence/post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

UTILIZATION OF THIRD PARTY LEGACY DATA LIST

the specification of which is attached hereto unless the following box is checked:

() was filed on _____ as US Application Serial No. or PCT International Application Number _____ and was amended on _____ (if applicable).

I hereby state that I have reviewed and understood the contents of the above-identified specification, including the claims, as amended by any amendment(s) referred to above. I acknowledge the duty to disclose all information which is material to patentability as defined in 37 CFR 1.56.

Foreign Application(s) and/or Claim of Foreign Priority

I hereby claim foreign priority benefits under Title 35, United States Code Section 119 of any foreign application(s) for patent or inventor(s) certificate listed below and have also identified below any foreign application for patent or inventor(s) certificate having a filing date before that of the application on which priority is claimed:

| COUNTRY | APPLICATION NUMBER | DATE FILED | PRIORITY CLAIMED UNDER 35 U.S.C. 119 |
|---------|--------------------|------------|--|
| | | | YES: <input type="checkbox"/> NO: <input type="checkbox"/> |
| | | | YES: <input type="checkbox"/> NO: <input type="checkbox"/> |

Provisional Application

I hereby claim the benefit under Title 35, United States Code Section 119(e) of any United States provisional application(s) listed below:

| APPLICATION SERIAL NUMBER | FILING DATE |
|---------------------------|-------------|
| | |
| | |

U.S. Priority Claim

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code Section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, Section 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

| APPLICATION SERIAL NUMBER | FILING DATE | STATUS (present/pending/abandoned) |
|---------------------------|-------------|------------------------------------|
| | | |
| | | |

POWER OF ATTORNEY:

As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith:

Customer Number 022879Place Customer
Number Bar Code
Label hereSend Correspondence to:
HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Direct Telephone Calls To:

Steven Orniston
(202) 396-4772

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Inventor: David BOHANCitizenship: USResidence: 2507 Haven Drive, Eagle, ID 83616Post Office Address: 2507 Haven Drive, Eagle, ID 83616David Bohan
Inventor's SignatureDate 10/2/2000